09/341633

Rec'd PCT/PTO 15 JUL 1999

09341633 071599

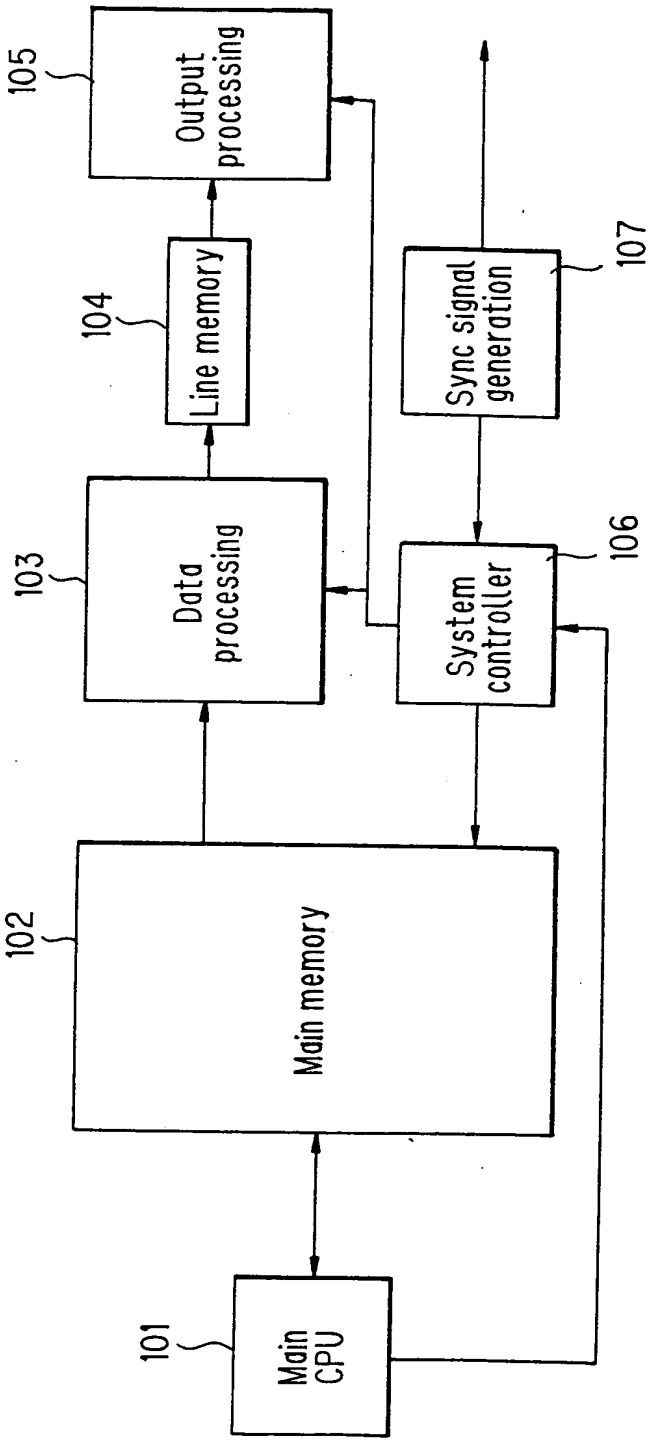


FIG. 1

FIG.2

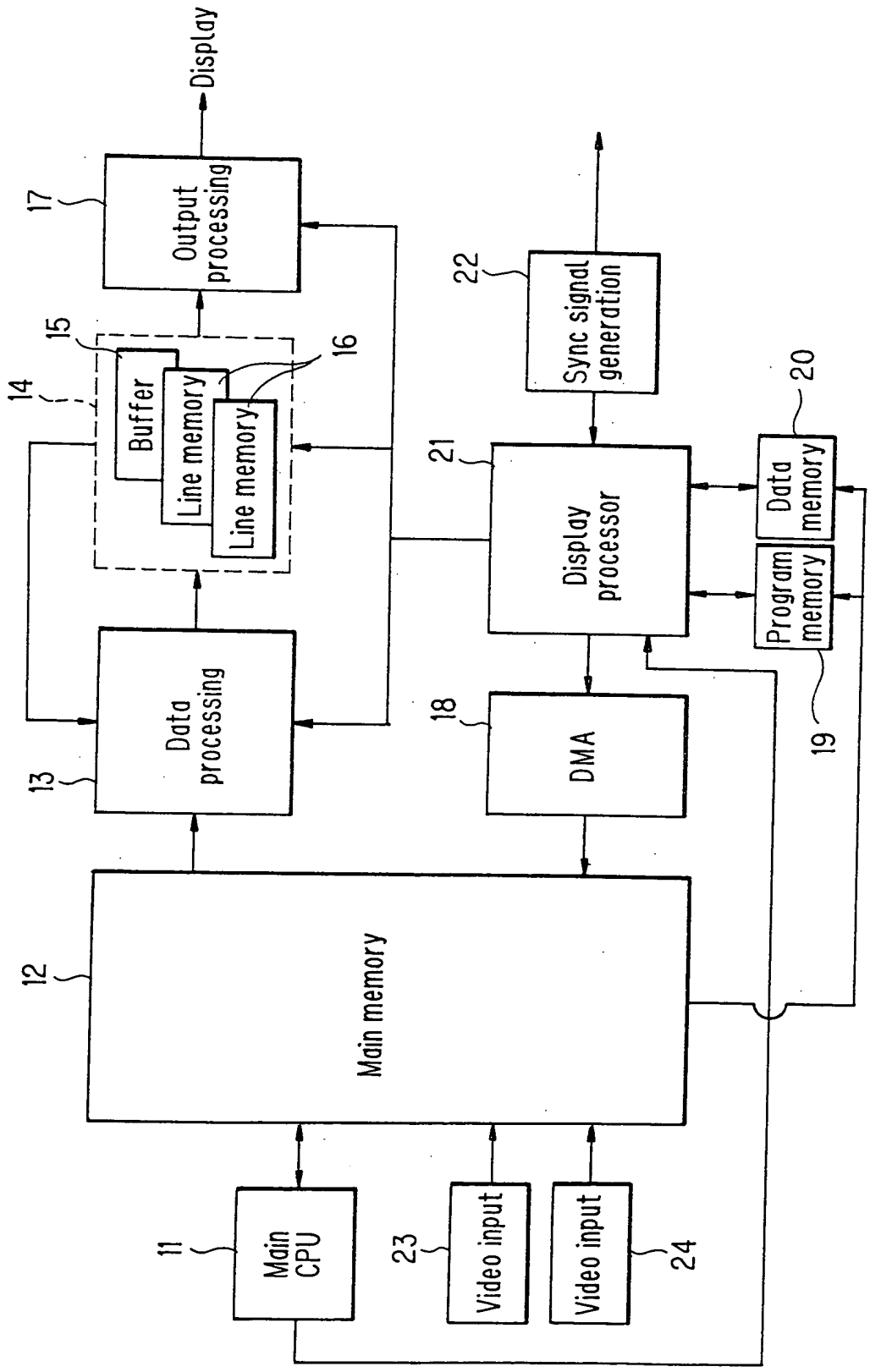

Display
17
Output processing
22
Sync signal generation
15
16
14
Buffer
Line memory
Line memory
20
21
Data memory
Display processor
Program memory
19
18
DMA
13
Data processing
12
Main memory
11
Main CPU
23
Video input
24
Video input

FIG.3

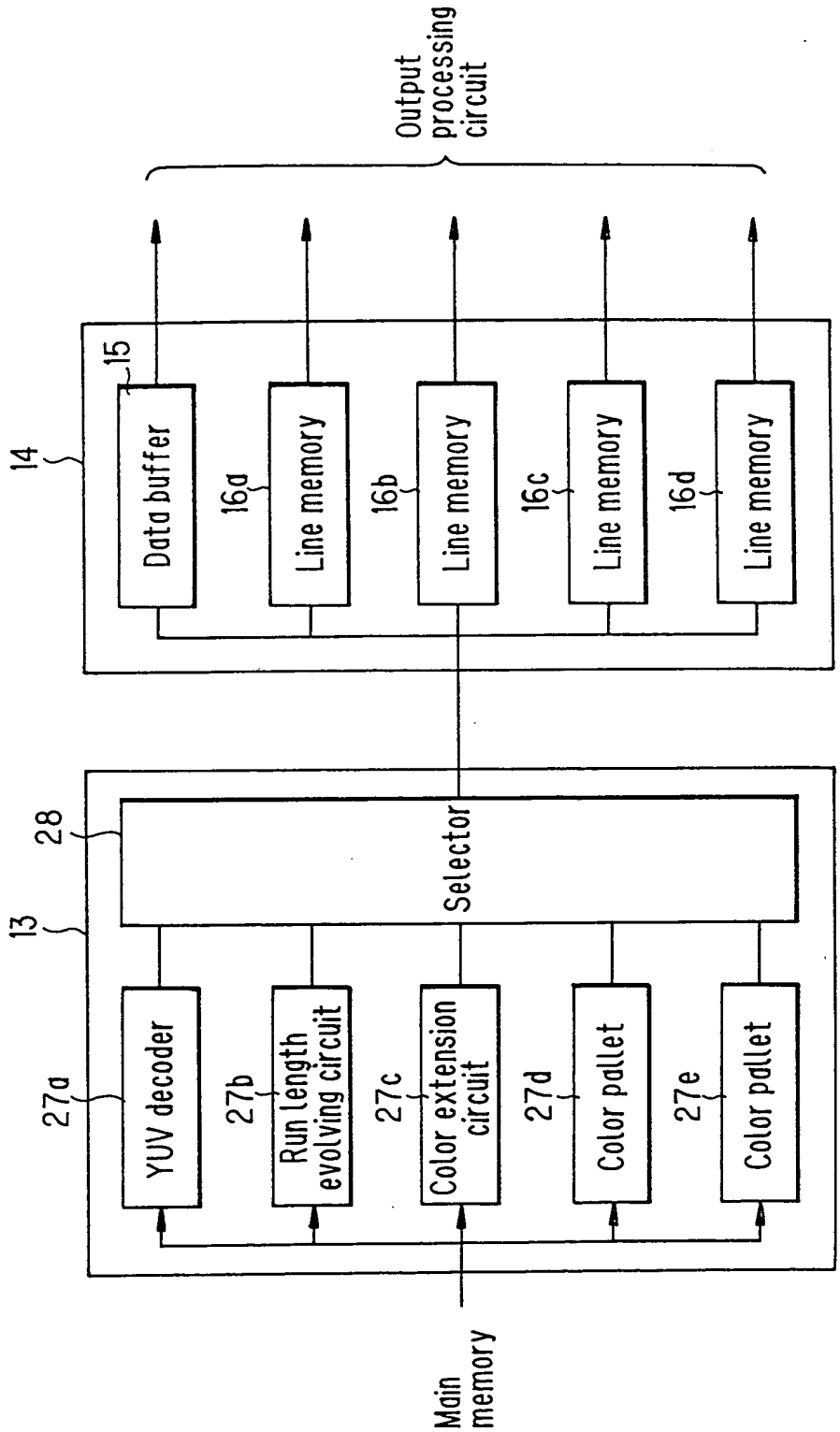
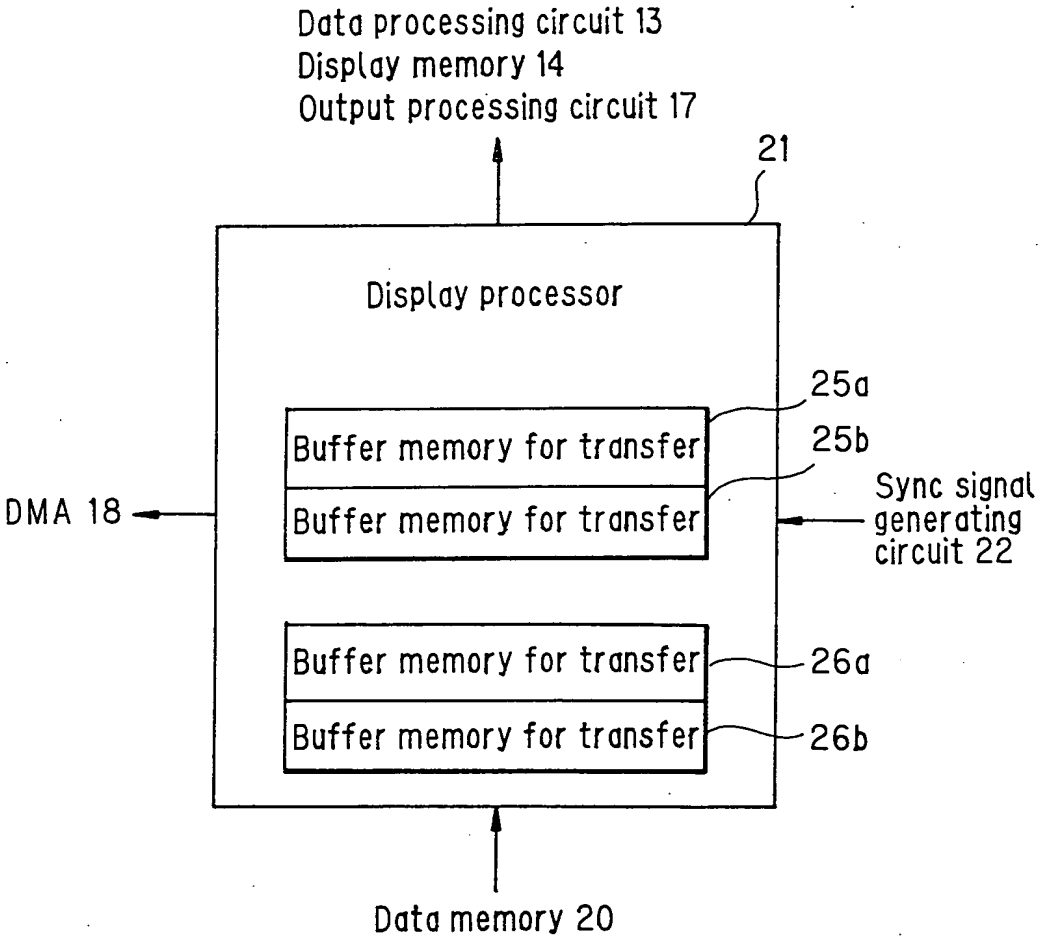

Output processing circuit
15
Data buffer
16a
Line memory
16b
Line memory
16c
Line memory
16d
Line memory
14
28
Selector
13
27a
YUV decoder
27b
Run length evolving circuit
27c
Color extension circuit
27d
Color pallet
27e
Color pallet
Main memory

FIG. 4


Data processing circuit 13
Display memory 14
Output processing circuit 17
21
Display processor
25a
Buffer memory for transfer
25b
Buffer memory for transfer
DMA 18
Sync signal generating circuit 22
Buffer memory for transfer
26a
Buffer memory for transfer
26b
Data memory 20

09/341633

FIG. 5A

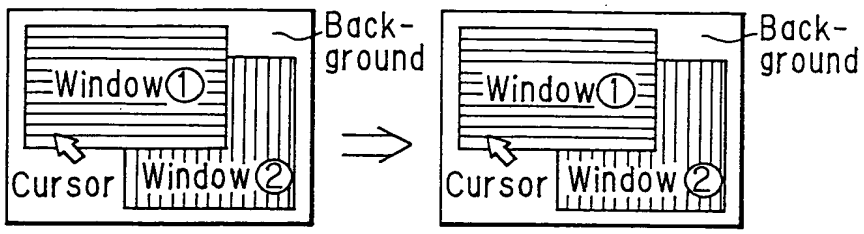
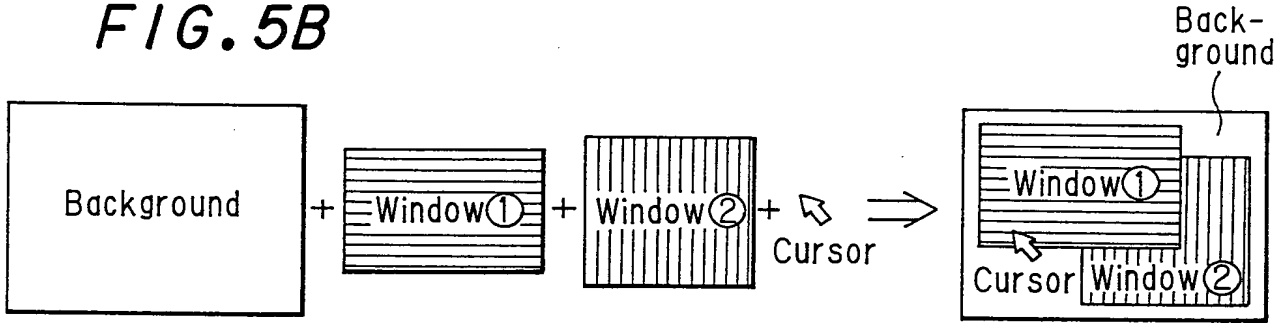


Image data already synthesized on the memory by the main processor

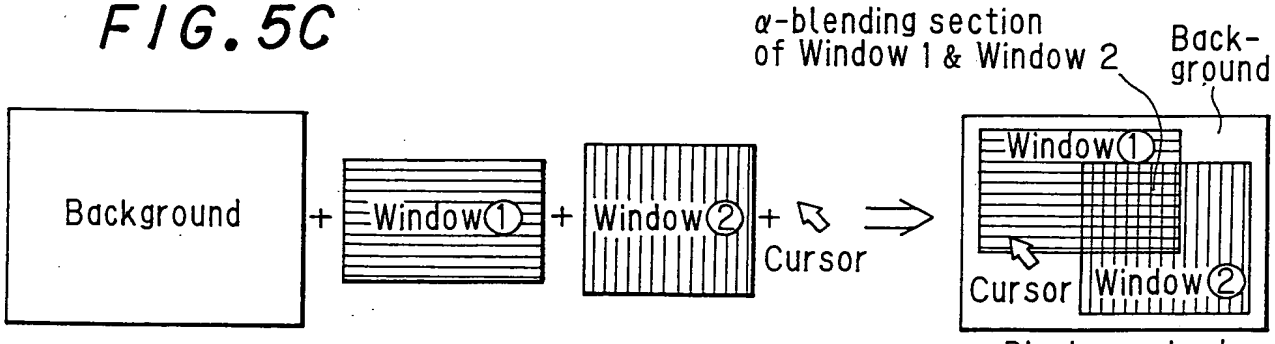
Display output

FIG. 5B



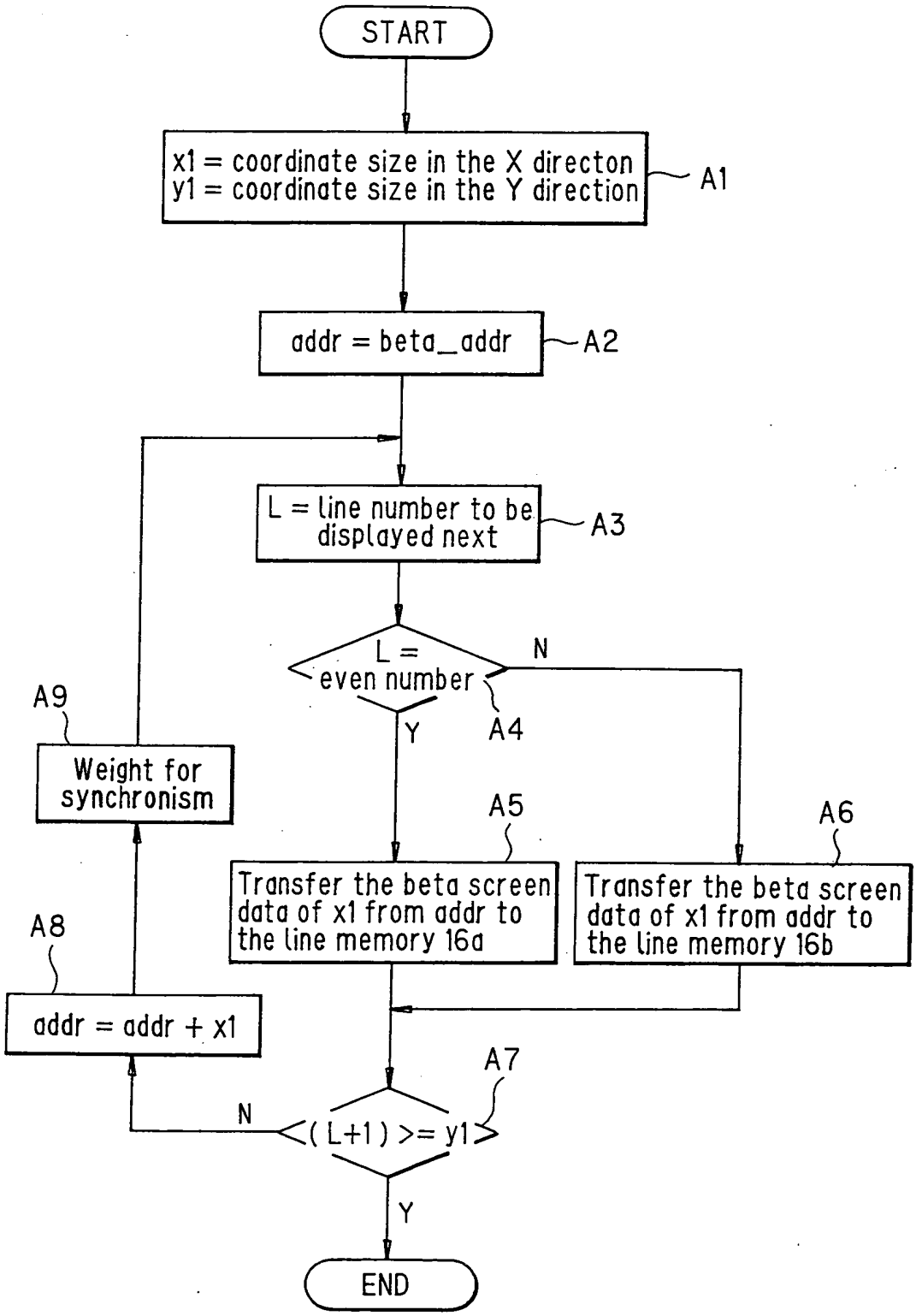
Display output

FIG. 5C



Display output

FIG. 6


START
x1 = coordinate size in the X directon
y1 = coordinate size in the Y direction
A1
addr = beta_addr
A2
L = line number to be displayed next
A3
L = even number
N
Y
A4
A9
Weight for synchronism
A5
Transfer the beta screen data of x1 from addr to the line memory 16a
A6
Transfer the beta screen data of x1 from addr to the line memory 16b
A8
addr = addr + x1
A7
N
(L+1) >= y1
Y
END

09/341633

FIG. 7

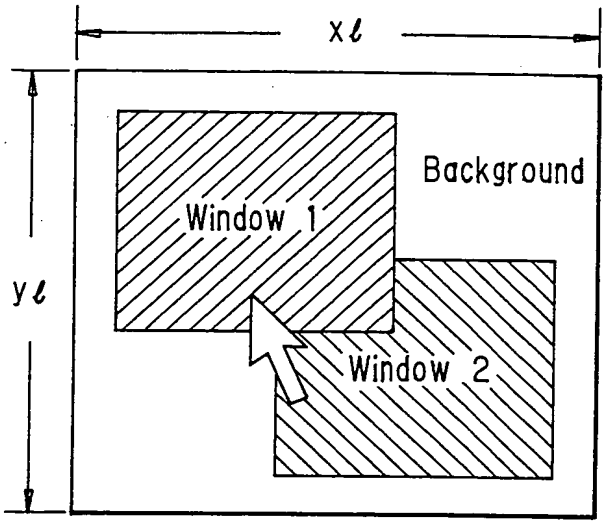
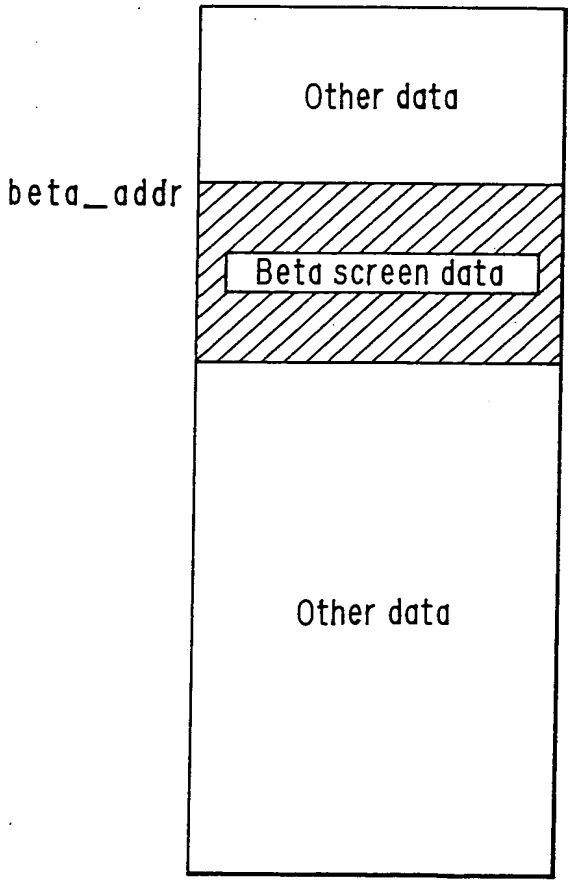


FIG. 8



WO 98/34365 PCT/US98/02256

FIG. 9

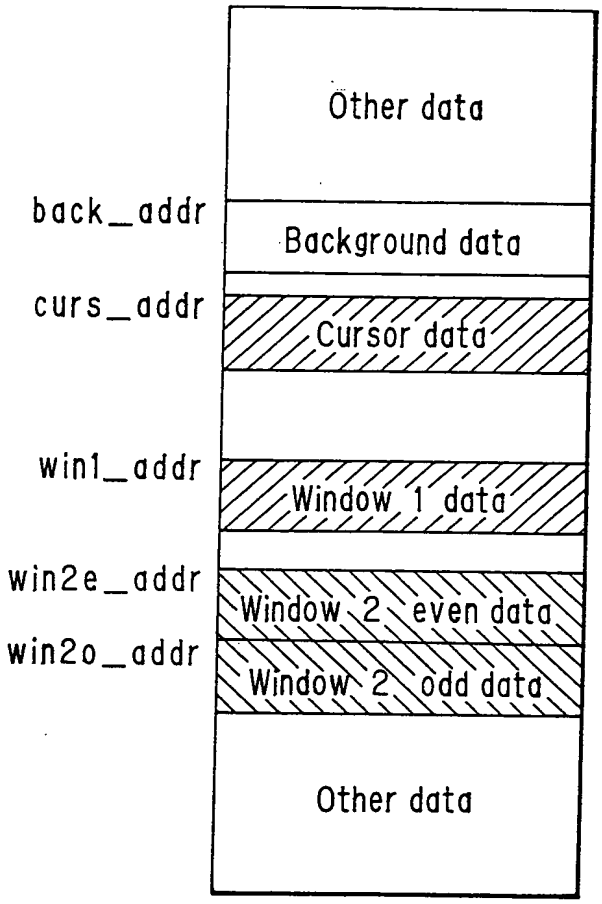

Other data
back_addr
Background data
curs_addr
Cursor data
win1_addr
Window 1 data
win2e_addr
Window 2 even data
win2o_addr
Window 2 odd data
Other data

FIG. 10

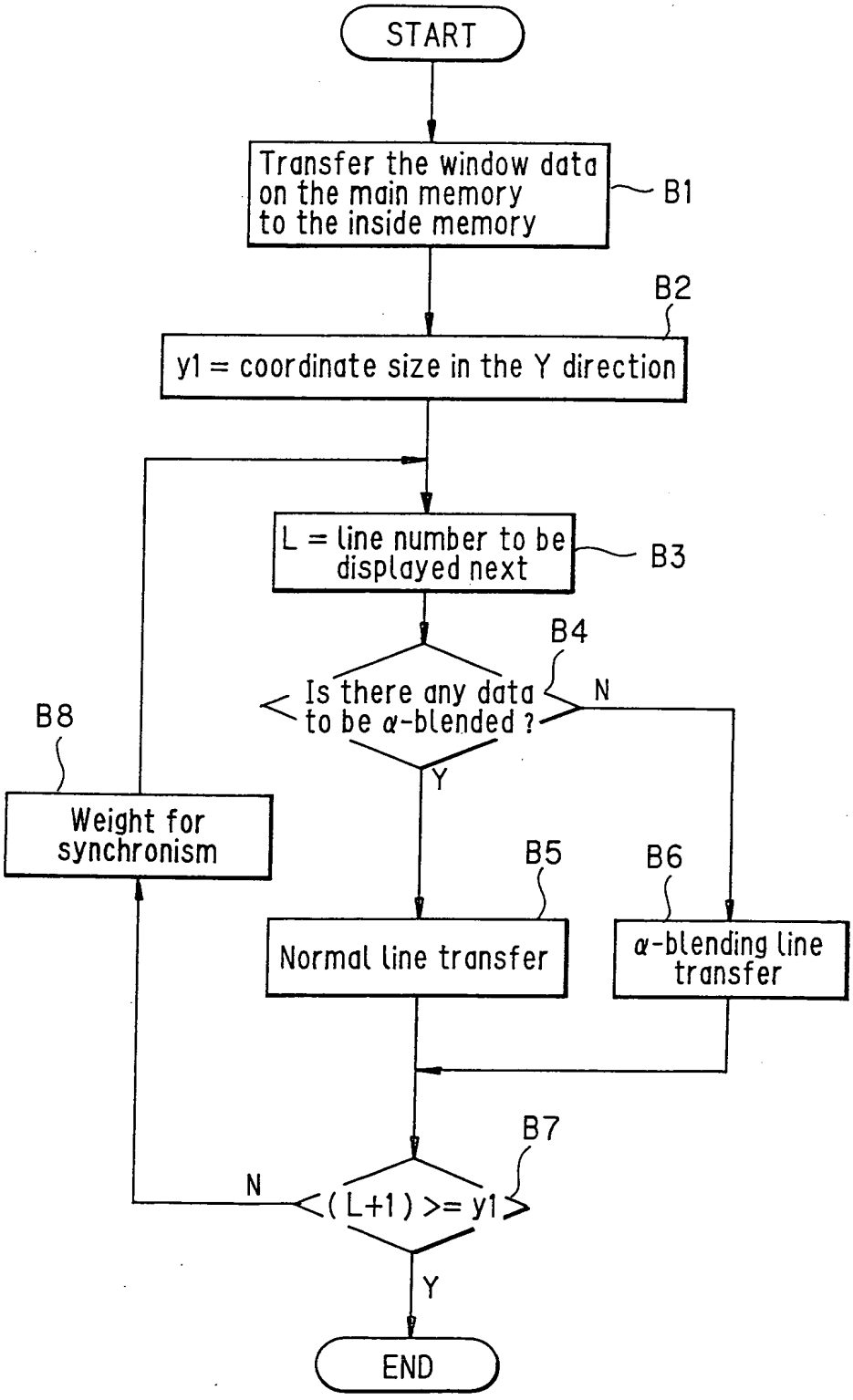

START
Transfer the window data on the main memory to the inside memory
B1
B2
y1 = coordinate size in the Y direction
L = line number to be displayed next
B3
B4
Is there any data to be α-blended ?
N
Y
B8
Weight for synchronism
B5
B6
Normal line transfer
α-blending line transfer
B7
N
(L+1) >= y1
Y
END

FIG. 11

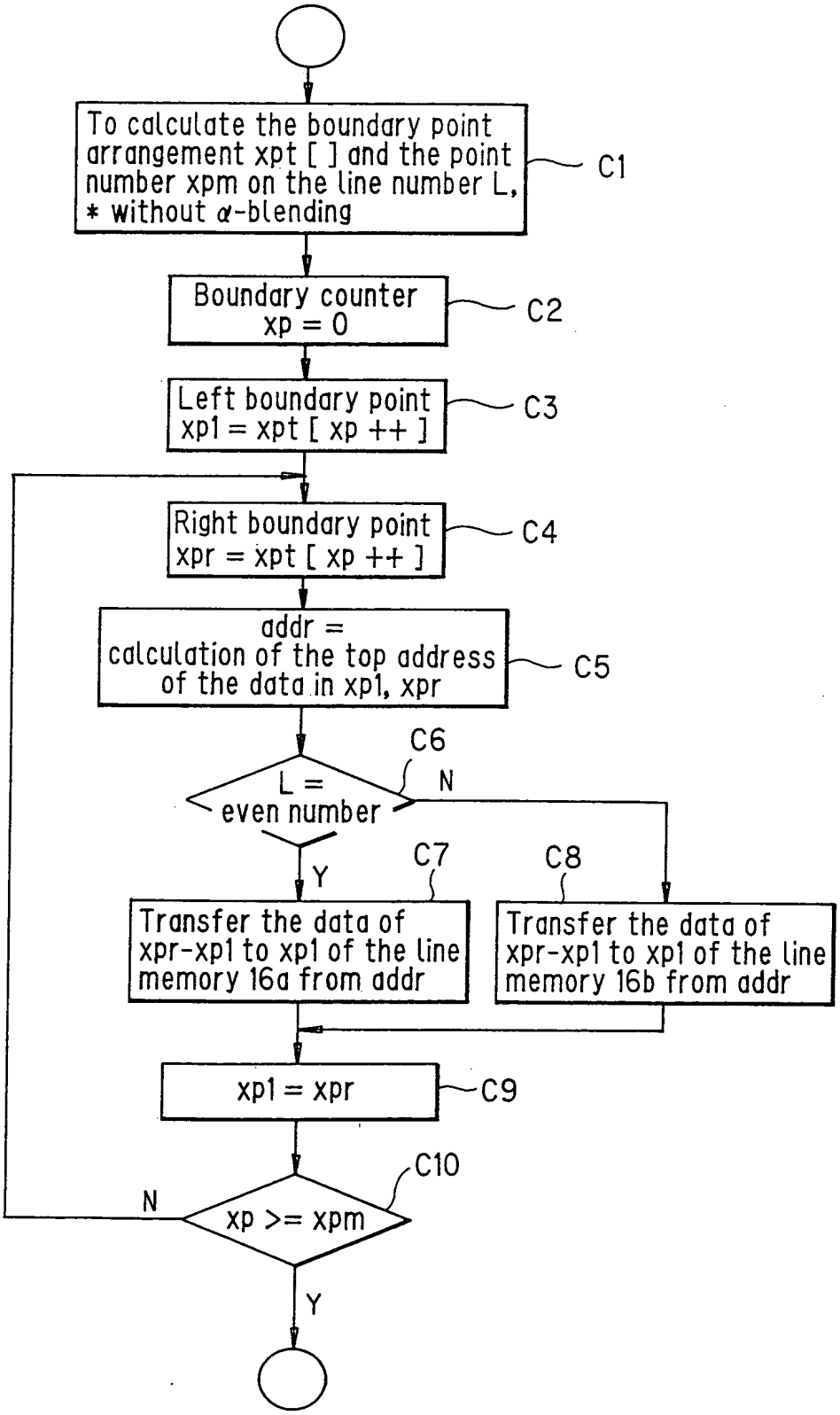

To calculate the boundary point arrangement xpt [ ] and the point number xpm on the line number L, * without α-blending
C1
Boundary counter xp = 0
C2
Left boundary point xp1 = xpt [xp ++]
C3
Right boundary point xpr = xpt [xp ++]
C4
addr = calculation of the top address of the data in xp1, xpr
C5
C6
L = even number
N
Y
C7
C8
Transfer the data of xpr-xp1 to xp1 of the line memory 16a from addr
Transfer the data of xpr-xp1 to xp1 of the line memory 16b from addr
xp1 = xpr
C9
C10
N
xp >= xpm
Y

FIG. 12A

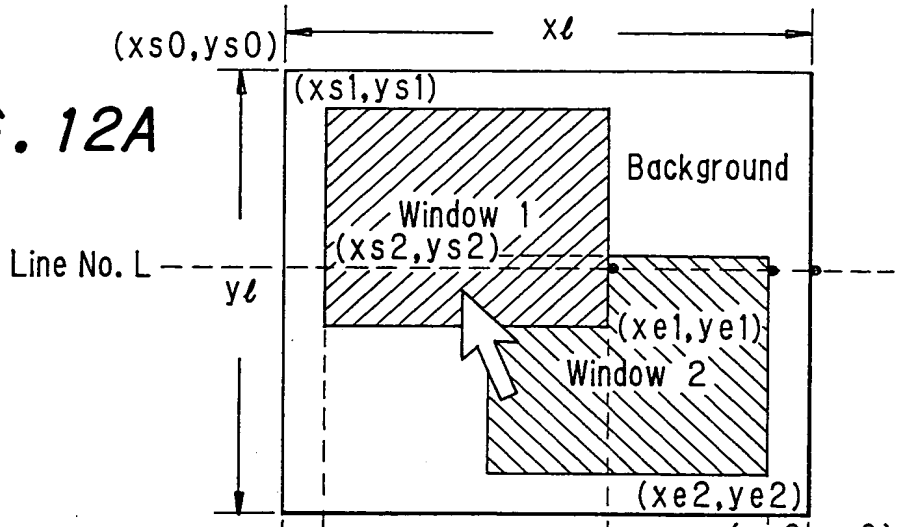


FIG. 12B

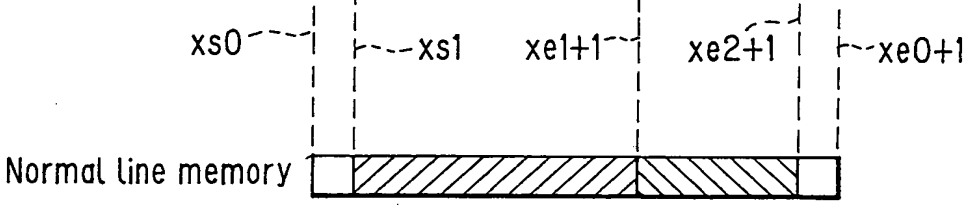


FIG. 13

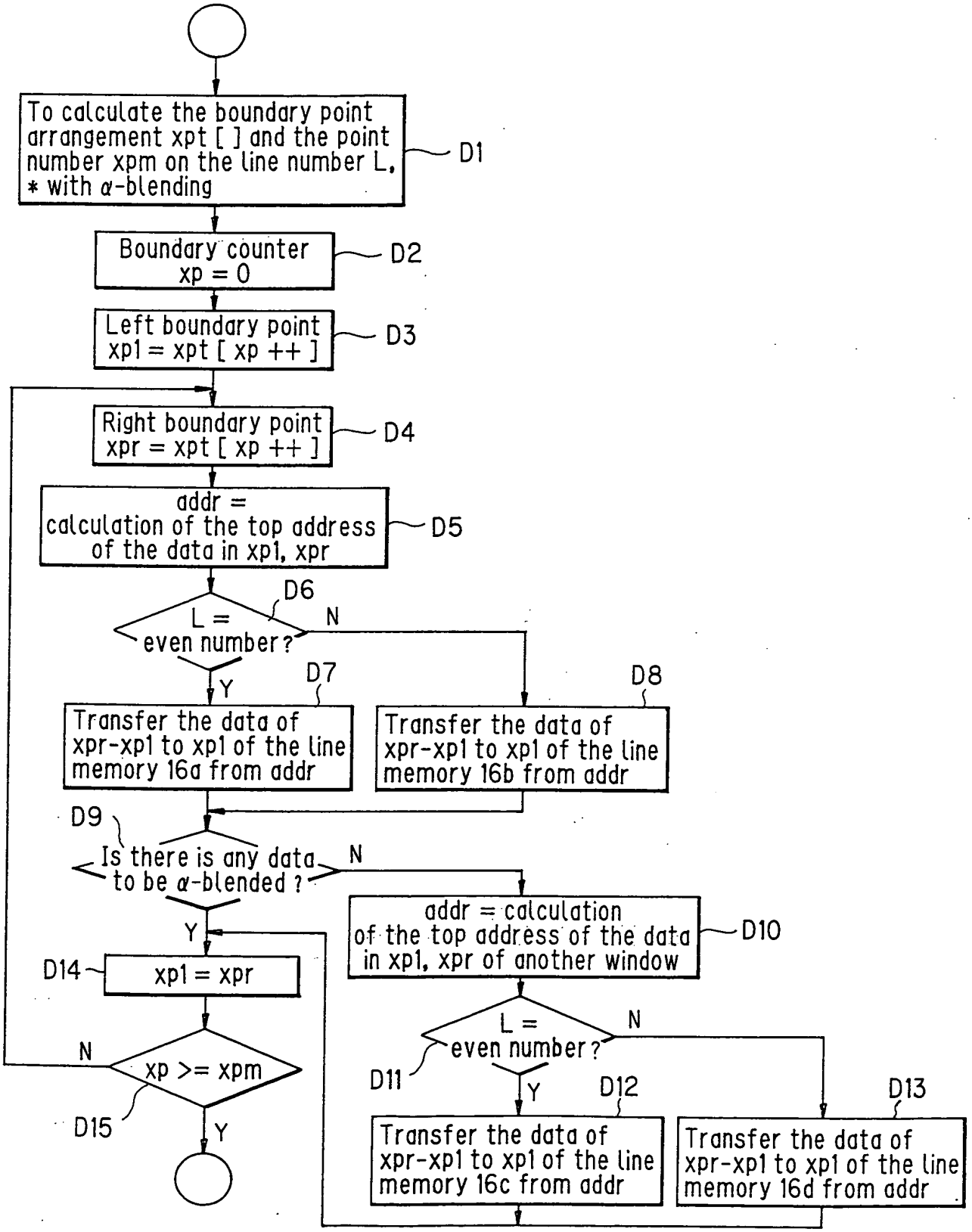

To calculate the boundary point arrangement xpt [ ] and the point number xpm on the line number L, * with α-blending
D1
Boundary counter xp = 0
D2
Left boundary point xp1 = xpt [xp ++]
D3
Right boundary point xpr = xpt [xp ++]
D4
addr = calculation of the top address of the data in xp1, xpr
D5
D6
L = even number?
N
Y
D7
Transfer the data of xpr-xp1 to xp1 of the line memory 16a from addr
D8
Transfer the data of xpr-xp1 to xp1 of the line memory 16b from addr
D9
Is there is any data to be α-blended ?
N
Y
addr = calculation of the top address of the data in xp1, xpr of another window
D10
D14
xp1 = xpr
N
xp >= xpm
D15
Y
L = even number?
N
D11
Y
D12
Transfer the data of xpr-xp1 to xp1 of the line memory 16c from addr
D13
Transfer the data of xpr-xp1 to xp1 of the line memory 16d from addr

FIG. 14A

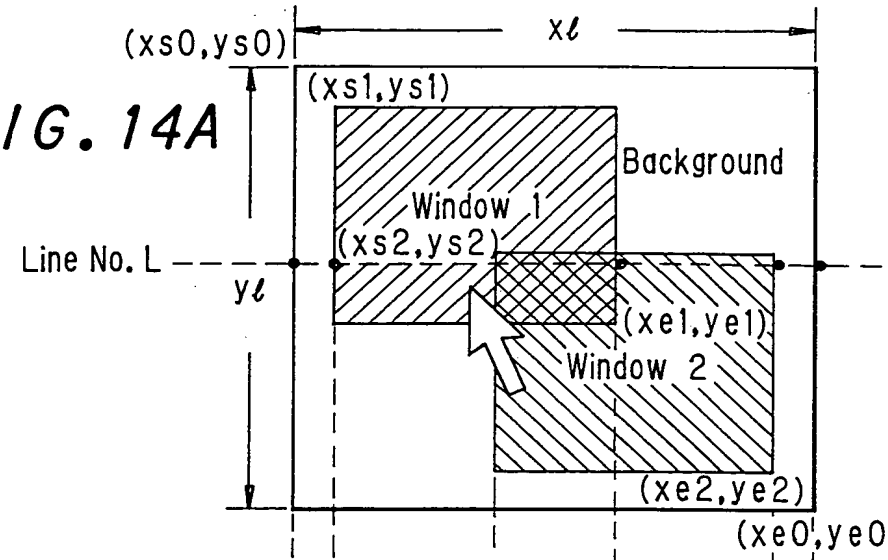


FIG. 14B

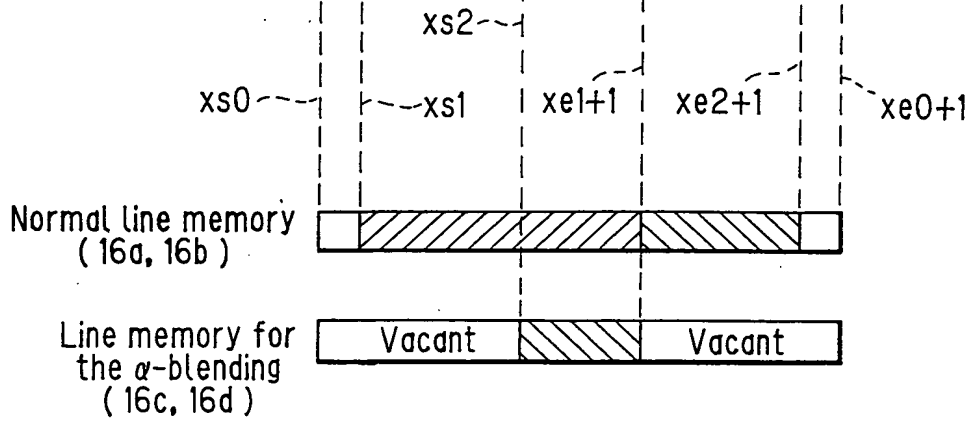


FIG. 15

Control data	State of motionr	Motion of each counter after processing		Write motion
		Readout counter	Write counter	
0 0	normal	+ 1	+ 1	to be done
0 1	contraction	+ 1	+ 0	not to be done
1 0	expansion	+ 0	+ 1	to be done
1 1	skip	+ 0	+ 1	not to be done

FIG. 16

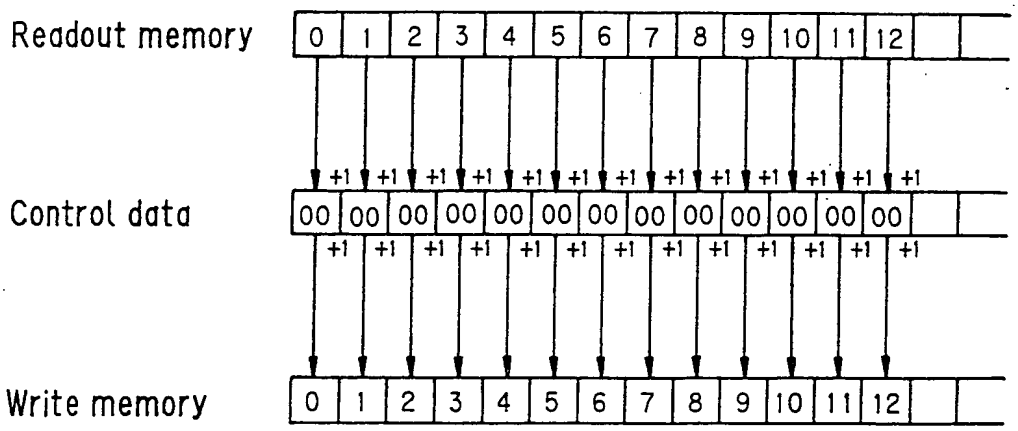


FIG. 17

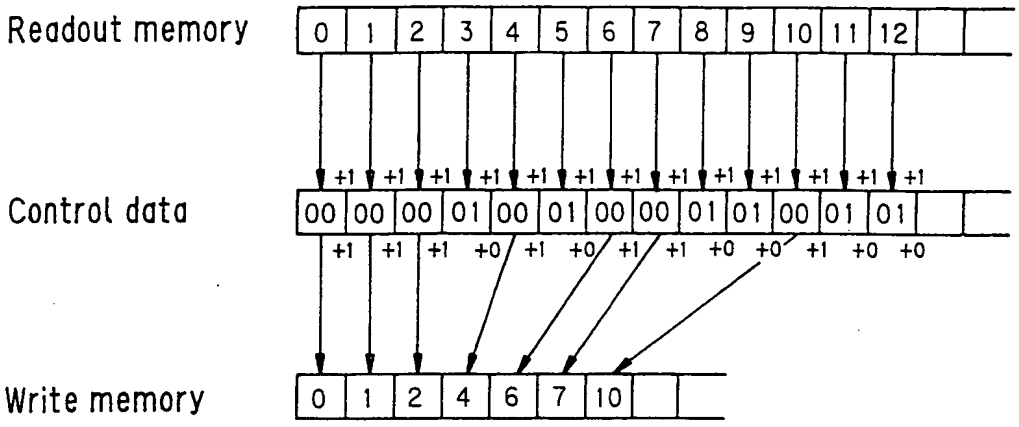

Readout memory
0 1 2 3 4 5 6 7 8 9 10 11 12
+1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1
Control data
00 00 00 01 00 01 00 00 01 01 00 01 01
+1 +1 +1 +0 +1 +0 +1 +1 +0 +0 +1 +0 +0
Write memory
0 1 2 4 6 7 10

FIG. 18

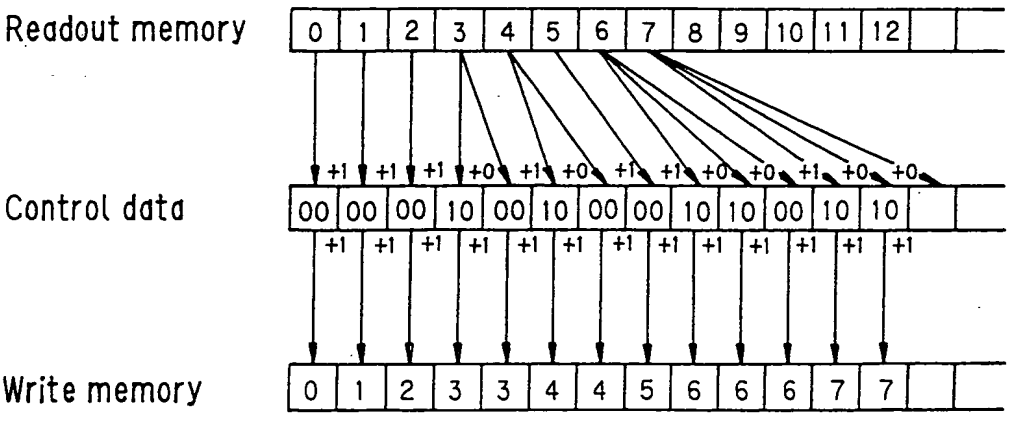

Readout memory
0 1 2 3 4 5 6 7 8 9 10 11 12
+1 +1 +1 +0 +1 +0 +1 +1 +0 +0 +1 +0 +0
Control data
00 00 00 10 00 10 00 00 10 10 00 10 10
+1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1
Write memory
0 1 2 3 3 4 4 5 6 6 6 7 7

FIG. 19

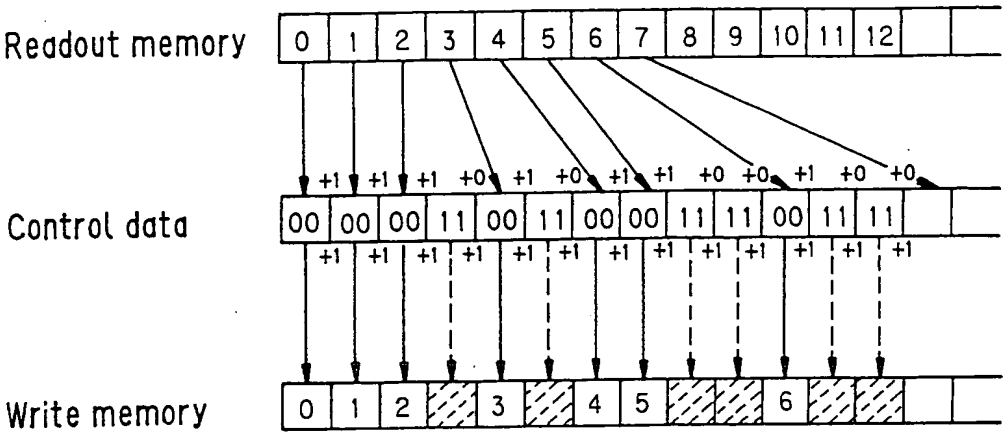

Readout memory
0 1 2 3 4 5 6 7 8 9 10 11 12
+1 +1 +1 +0 +1 +0 +1 +1 +0 +0 +1 +0 +0
Control data
00 00 00 11 00 11 00 00 11 11 00 11 11
+1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1
Write memory
0 1 2 3 4 5 6

FIG. 20

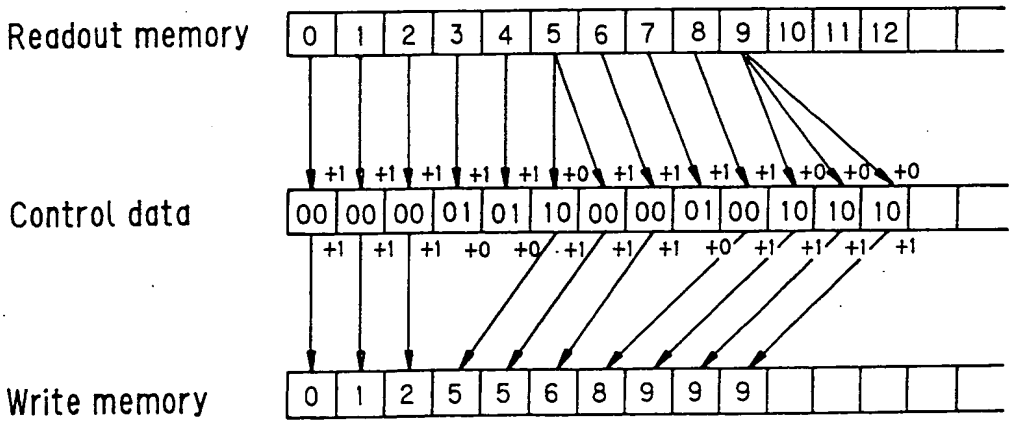

Readout memory
0 1 2 3 4 5 6 7 8 9 10 11 12
+1 +1 +1 +1 +1 +1 +0 +1 +1 +1 +1 +0 +0 +0
Control data
00 00 00 01 01 10 00 00 01 00 10 10 10
+1 +1 +1 +0 +0 +1 +1 +1 +0 +1 +1 +1 +1
Write memory
0 1 2 5 5 6 8 9 9 9

FIG. 21

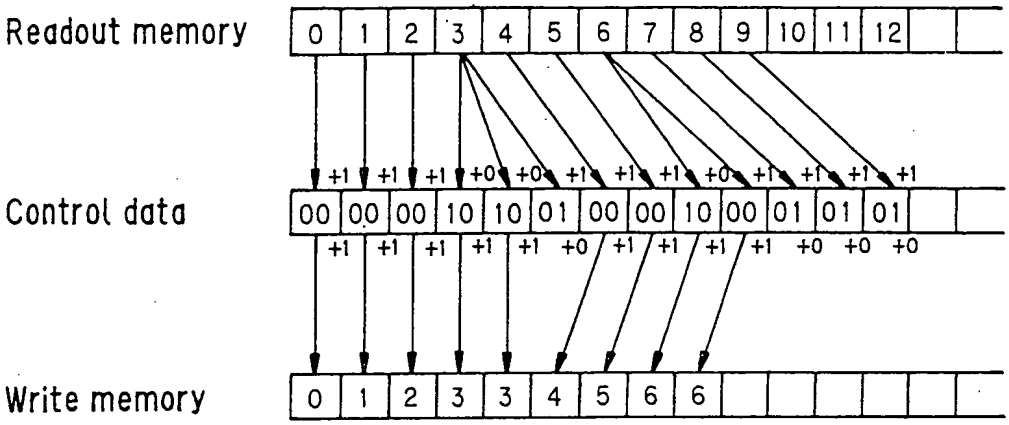

Readout memory
0 1 2 3 4 5 6 7 8 9 10 11 12
+1 +1 +1 +0 +0 +1 +1 +1 +0 +1 +1 +1 +1
Control data
00 00 00 10 10 01 00 00 10 00 01 01 01
+1 +1 +1 +1 +1 +0 +1 +1 +1 +1 +0 +0 +0
Write memory
0 1 2 3 3 4 5 6 6

FIG. 22

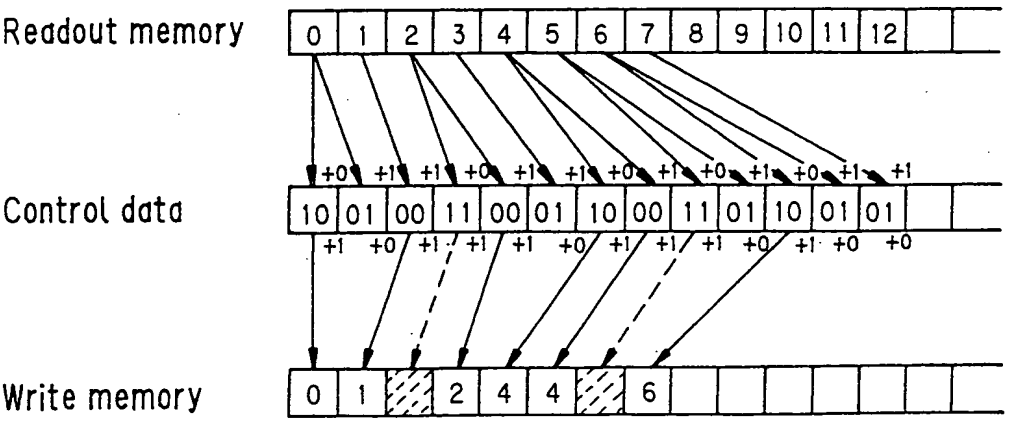

Readout memory
0 1 2 3 4 5 6 7 8 9 10 11 12
+0 +1 +1 +0 +1 +1 +0 +1 +0 +1 +0 +1 +1
Control data
10 01 00 11 00 01 10 00 11 01 10 01 01
+1 +0 +1 +1 +1 +0 +1 +1 +1 +0 +1 +0 +0
Write memory
0 1 2 4 4 6

FIG. 23

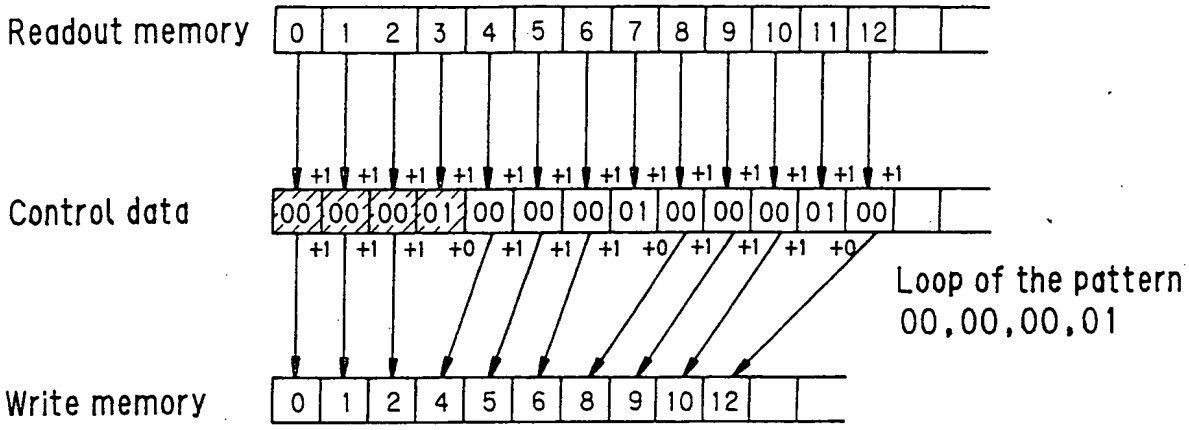
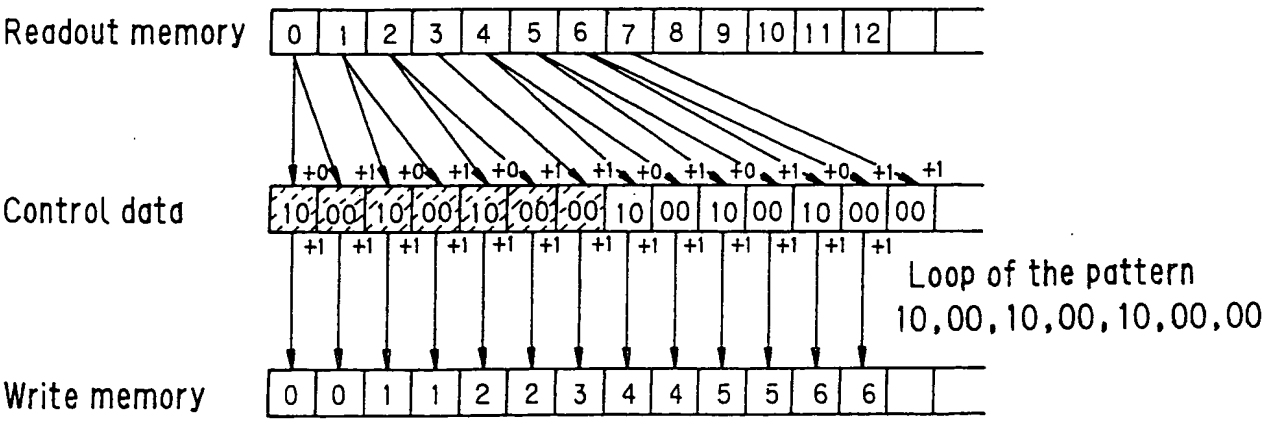

Readout memory
0 1 2 3 4 5 6 7 8 9 10 11 12
+1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1
Control data
00 00 00 01 00 00 00 01 00 00 00 01 00
+1 +1 +1 +0 +1 +1 +1 +0 +1 +1 +1 +0
Loop of the pattern
00,00,00,01
Write memory
0 1 2 4 5 6 8 9 10 12

FIG. 24


Readout memory
0 1 2 3 4 5 6 7 8 9 10 11 12
+0 +1 +0 +1 +0 +1 +1 +0 +1 +0 +1 +0 +1 +1
Control data
10 00 10 00 10 00 00 10 00 10 00 10 00 00
+1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1 +1
Loop of the pattern
10,00,10,00,10,00,00
Write memory
0 0 1 1 2 2 3 4 4 5 5 6 6

RECEIVED 15 JUL

FIG. 25

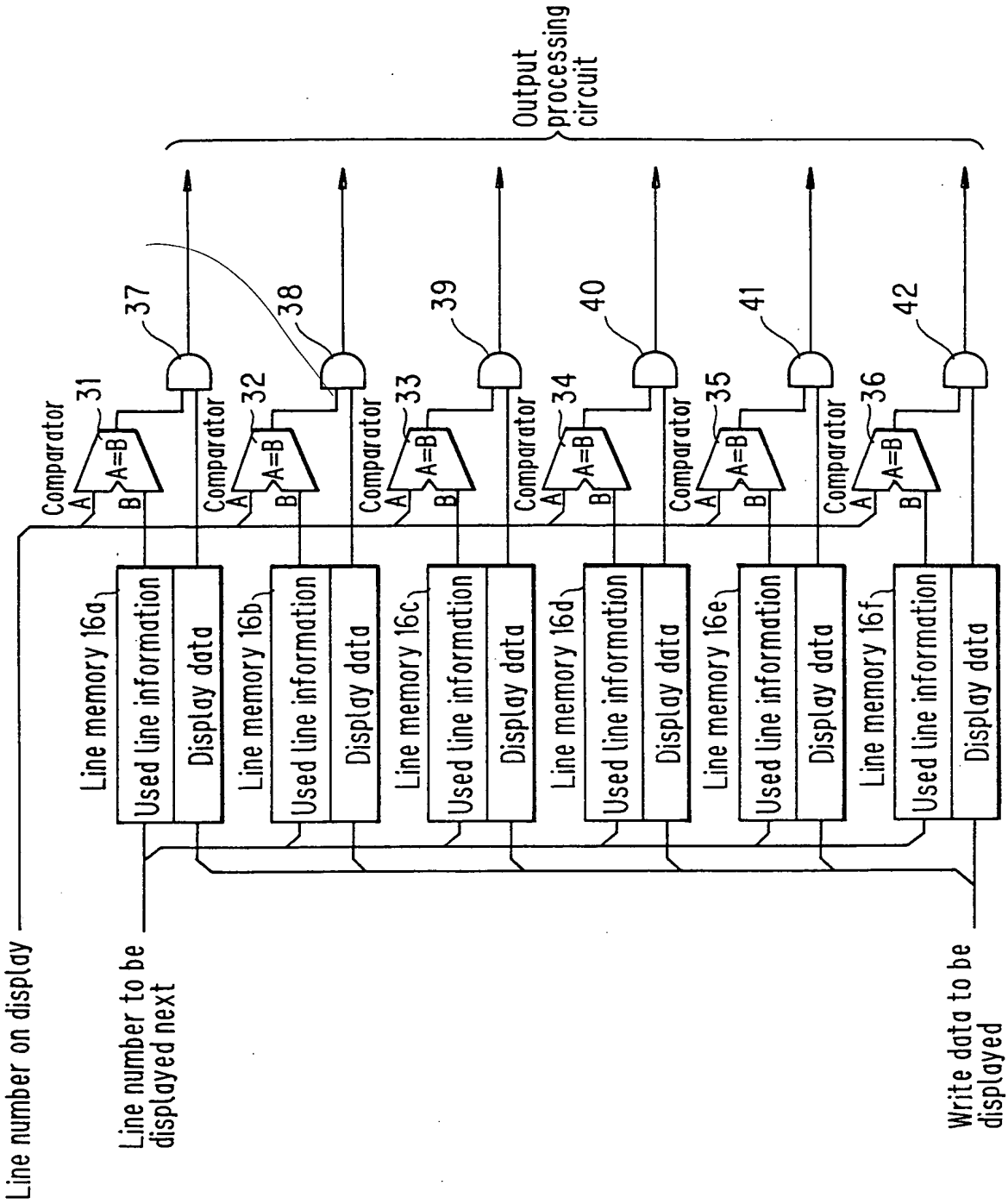


FIG. 26A

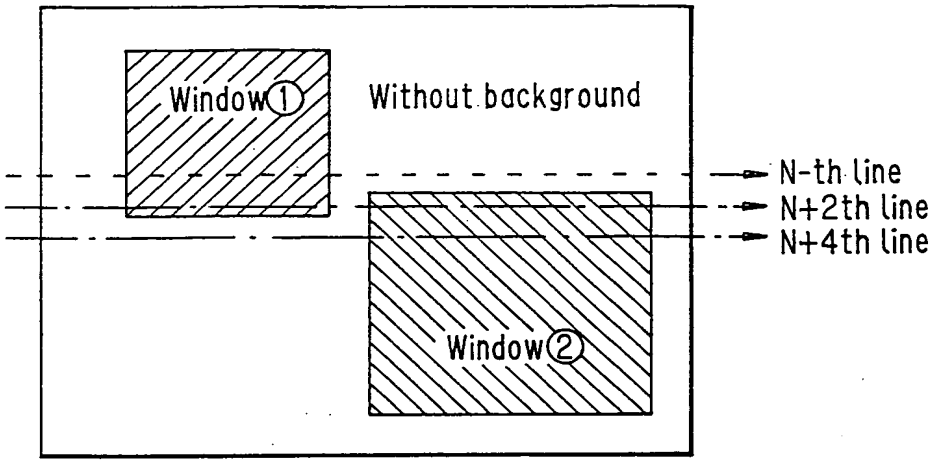


FIG. 26B

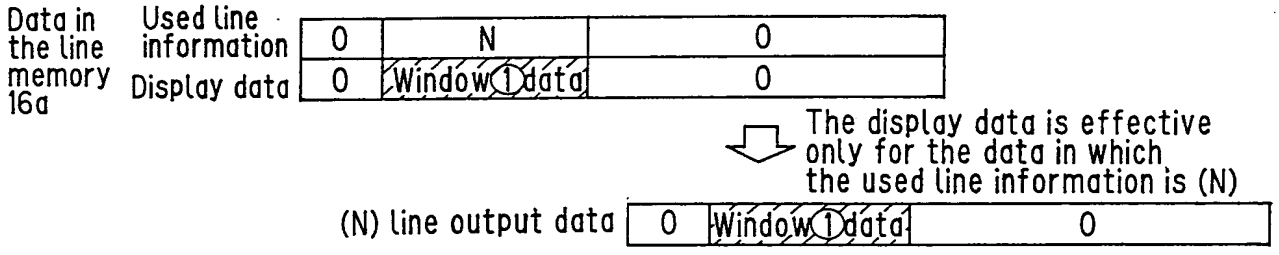


FIG. 26C

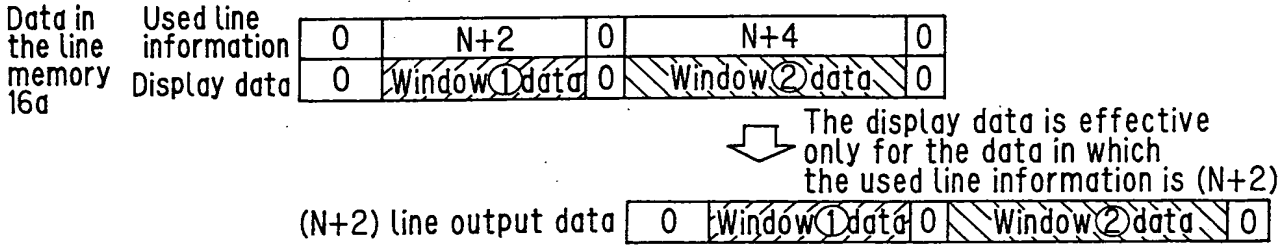
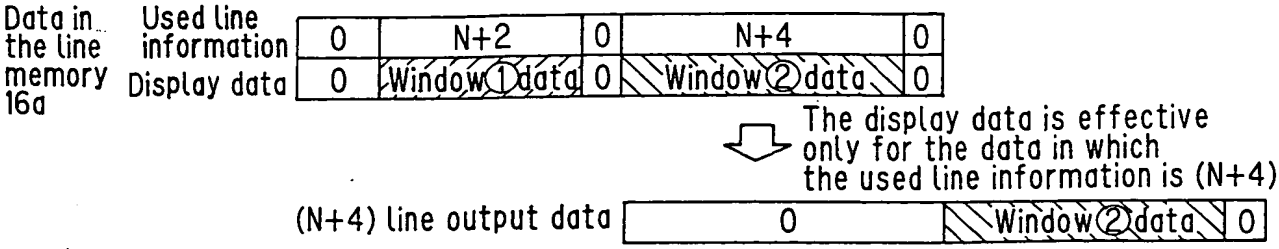
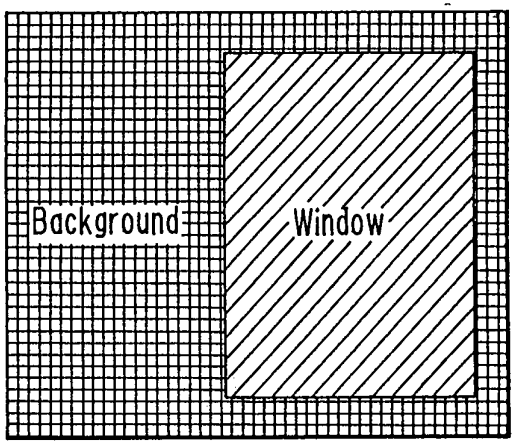


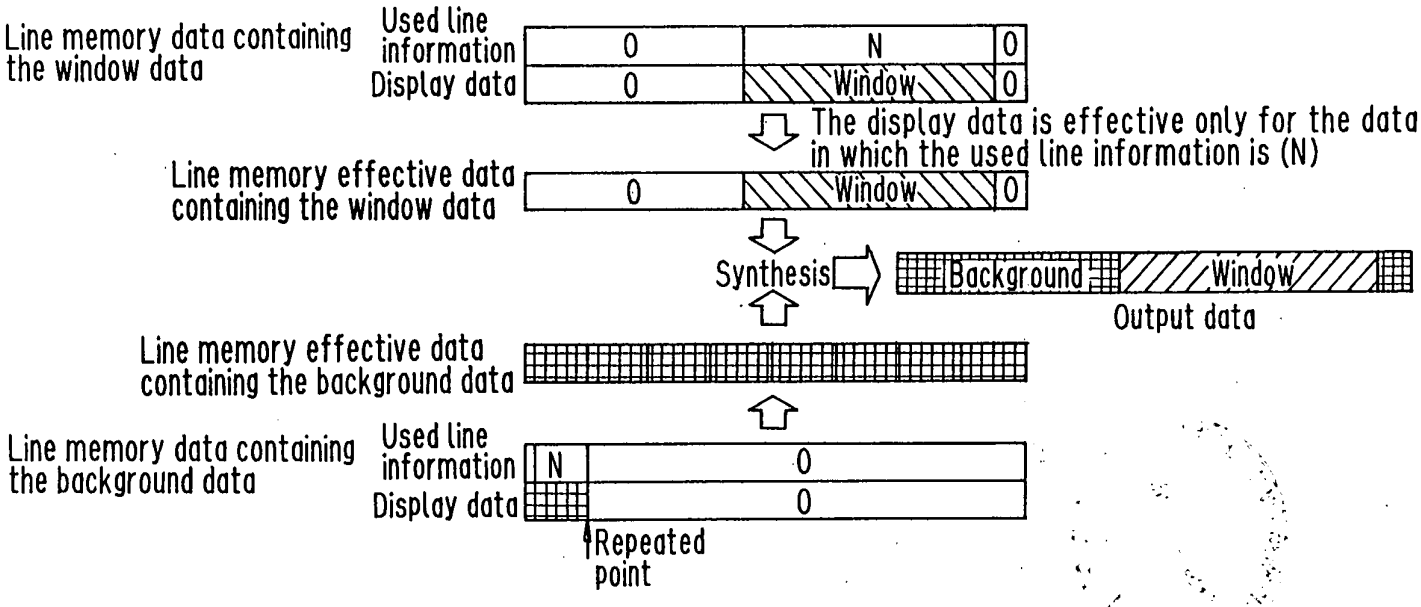
FIG. 26D



66ST20 EE9TH60

FIG.27





Rec'd Jul. 31. 01 07:10pm 09/341633